

1 4. Claims:

2 (1) A message processing apparatus characterized by having:

3 process requestor search information management means for managing process requestor
4 search information for searching for an applicable process requestor as to an agent start
5 cause event;

6 acceptance means for accepting the agent start cause event;

7 list information creation means for, based on said process requestor search information,
8 creating list information on process requestors to which a message generated due to said
9 agent start cause event is applied;

10 a plurality of agents associated with the process requestors, stored in a persistent storage,
11 readable from the persistent storage to a cache memory as a program execution area and
12 discardable from the cache memory, each agent operating only when existing in the cache
13 memory to be able to process the message in a message queue corresponding to the agent;

14 insertion and reading means for, of the process requestors included in said list
15 information, selecting a plurality of unselected ones as the process requestors to be
16 inserted and read, inserting said message into the message queues related to the process
17 requestors to be inserted and read and reading the agents related to said process requestors
18 from the persistent storage to the cache memory;

19 agent instruction means for instructing the agent related to the message queue having the
20 message inserted therein to operate; and

1 repetitive instruction means for, in the case where the unselected one remains among the
2 process requestors included in said list information, waiting for termination of the process
3 of all the agents in operation and instructing said insertion and reading means to repeat
4 the process.

5 (2) The message processing apparatus according to claim 1, characterized in that each
6 agent sends a notice to the process requestor associated with the agent as to the message
7 in the message queue associated with the agent.

8 (3) A message processing apparatus characterized by having:

9 process requestor search information management means for managing process requestor
10 search information for searching for an applicable process requestor as to an agent start
11 cause event;

12 acceptance means for accepting the agent start cause event;

13 list information creation means for, based on said process requestor search information,
14 creating list information on process requestors to which a message generated due to said
15 agent start cause event is applied;

16 a plurality of agents associated with the process requestors, stored in a persistent storage,
17 readable from the persistent storage to a cache memory as a program execution area and
18 abandonable from the cache memory, each agent operating only when existing in the
19 cache memory to be able to process the message in a message queue corresponding to the
20 agent;

1 a first message queue processing mechanism;

2 a second message queue processing mechanism;

3 selection means for selecting either one of the first and second message queue processing

4 mechanisms; and

5 agent instruction means for, as to the agents related to the message queue having the

6 message inserted therein, immediately instructing the agent to operate if the agent is in

7 the cache memory, and reading the agent from said persistent storage to said cache

8 memory and then instructing the agent to operate if the agent is not in the cache memory;

9 and wherein:

10 the first message queue processing mechanism has insertion means for inserting said

11 message into the message queues related to all the process requestors included in said list

12 information; and

13 the second message queue processing mechanism has:

14 insertion and reading means for, of the process requestors included in said list

15 information, selecting a plurality of unselected ones as the process requestors to be

16 inserted and read, inserting said message into the message queues related to the process

17 requestors to be inserted and read and reading the agents related to said process requestors

18 from the persistent storage to the cache memory; and

19 repetitive instruction means for, in the case where the unselected one remains among the

20 process requestors included in said list information, waiting for termination of the process

1 of all the agents in operation and instructing said insertion and reading means to repeat
2 the process.

3 (4) The message processing apparatus according to claim 3, characterized in that
4 selection of said selection means is based on an instruction of an operator.

5 (5) The message processing apparatus according to claim 3, characterized in that
6 selection of said selection means is based on the estimated number of the process
7 requestors to which the message generated due to the agent start cause event of this time
8 is applied, estimated hit rate in the cache memory as to the agents related to the process
9 requestors to which the message generated due to the agent start cause event of this time
10 is applied, estimated work time, in the case where said selection means selects the first
11 message queue processing mechanism, from acceptance of the information by said
12 acceptance means until obtaining the list information on the process requestors to which
13 the message is applied and inserting the message into the message queues of all the
14 agents, estimated work time, in the case where said selection means selects the second
15 message queue processing mechanism, from acceptance of the information by said
16 acceptance means until obtaining the list information on the process requestors to which
17 the message is applied and inserting the message into the message queues of all the
18 agents, estimated time from determination of use as to the agent in the cache memory
19 until completion of the process by the determined agent, and/or estimated time from
20 determination of use as to the agent outside the cache memory until the completion of the
21 process by the determined agent.

22 (6) A message processing apparatus characterized by having:

23 process requestor search information management means for managing process requestor

1 search information for searching for an applicable process requestor as to an agent start
2 cause event;

3 acceptance means for accepting the agent start cause event;

4 process requestor determination means for determining the process requestor to which the
5 message generated due to said agent start cause event is applied based on said process
6 requestor search information;

7 a plurality of agents associated with the process requestors, stored in a persistent storage,
8 readable from the persistent storage to a cache memory as a program execution area and
9 abandonable from the cache memory, each agent becoming operable when existing in the
10 cache memory;

11 at least one sub-process priority determination means for determining process priority
12 about each message as sub-process priority based on a single standard of value;

13 compound process priority determination means for, when the total number of said
14 sub-process priority determination means is two or more, determining compound process
15 priority about each message based on the sub-process priority individually determined as
16 to each message by each sub-process priority determination means, and when the total
17 number of said sub-process priority determination means is one, determining as the
18 compound process priority the sub-process priority determined by said one sub-process
19 priority determination means as to each message; and

20 agent instruction means for rendering the message of the highest compound process
21 priority among the messages held by each message queue as the message of the highest

1 priority, and between the agents related to the message queues of which compound
2 process priority of the message of the highest priority is the same, instructing the agent
3 existing in the cache memory to operate in preference to the agent not existing therein.

4 (7) The message processing apparatus according to claim 6, characterized in that said
5 standard of value may be related to the contents of the message or to the process
6 requestors to which the message is applied.

7 (8) The message processing apparatus according to claim 7, characterized in that the
8 predetermined standard of value related to the contents of the message includes the
9 standard of value related to emergency of processing of the message.

10 (9) The message processing apparatus according to claim 7, characterized in that the
11 standard of value related to the process requestors to which the message is applied
12 includes the standard of value related to rating of the process requestors.

13 (10) The message processing apparatus according to claim 6, characterized in that, when
14 there are a plurality of messages held by the message queue related to the agent which
15 once started the process by said agent instruction means, the agent continuously processes
16 all those messages or the messages of which compound process priority is within a
17 predetermined position in descending rank.

18 (11) The message processing apparatus according to claim 6, characterized in that:

19 there is agent management means including said sub-process priority determination
20 means and said compound process priority determination means;

21 said agent management means has existence detection means for detecting whether or not

1 each agent exists in the persistent storage, grouping information management means for
2 grouping the agents and managing grouping information based on determination results
3 of the existence detection means and the compound process priority of each agent, and
4 update instruction means for instructing the grouping information management means to
5 update the grouping information; and

6 said agent instruction means instructs the agents to operate in order based on the grouping
7 information of said agent management means.

8 (12) A message processing apparatus characterized by having:

9 process requestor search information management means for managing process requestor
10 search information for searching for an applicable process requestor as to an agent start
11 cause event;

12 acceptance means for accepting the agent start cause event;

13 acceptance order information management means for managing acceptance order
14 information on the agent start cause events accepted by said acceptance means;

15 a plurality of agents associated with the process requestors, stored in a persistent storage,
16 readable from the persistent storage to a cache memory as a program execution area and
17 abandonable from the cache memory, each agent operating only when existing in the
18 cache memory to be able to process the message in a message queue corresponding to the
19 agent;

20 a plurality of threads mutually operable in parallel, each thread detecting the process

1 requestors to which the message generated due to said agent start cause event is applied
2 based on said process requestor search information and inserting said message into the
3 message queues related to the process requestors;

4 allocation means for allocating to each thread the agent start cause event to be processed
5 by that thread;

6 proceeding information management means for managing proceeding information on the
7 process by the thread as to each agent start cause event accepted by said acceptance
8 means;

9 determination means for, as to the agent start cause event of which process proceeding
10 information is the information on thread process termination (hereafter, referred to as a
11 “determined agent start cause event”), determining whether or not, of the agent start cause
12 events accepted by said acceptance means prior to the determined agent start cause event,
13 there is any agent start cause event of which thread process is unfinished; and

14 agent control means for controlling the process by the agent as to the message generated
15 due to the determined agent start cause event determined as “yes” by said determination
16 means.

17 (13) The message processing apparatus according to claim 12, characterized by having
18 said agent control means for allowing the process by the agent as to the message
19 generated due to the determined agent start cause event determined as “no” by said
20 determination means.

21 (14) The message processing apparatus according to claim 13, characterized by having

1 said determination means for, in the case where the agent start cause event immediately
2 following the agent start cause event determined as “no” in acceptance order is already
3 determined as “yes,” changing the determination result from “yes” to “no.”

4 (15) The message processing apparatus according to claim 14, characterized by having
5 said agent for, in the case of processing the message queue in which the messages
6 generated due to a plurality of agent start cause events of which determination result by
7 said determination means is “no” continue in an acceptance order direction, continuously
8 processing the plurality of continuous messages.

9 (16) A message processing method characterized by having:

10 a process requestor search information management step of managing process requestor
11 search information for searching for an applicable process requestor as to an agent start
12 cause event;

13 an acceptance step of accepting the agent start cause event;

14 a list information creation step of, based on said process requestor search information,
15 creating list information on process requestors to which a message generated due to said
16 agent start cause event is applied;

17 an agent setting step of setting a plurality of agents, the agents being associated with the
18 process requestors, stored in a persistent storage, readable from the persistent storage to a
19 cache memory as a program execution area and abandonable from the cache memory,
20 each agent operating only when existing in the cache memory to be able to process the
21 message in a message queue corresponding to the agent;

1 an insertion and reading step of, of the process requestors included in said list
2 information, selecting a plurality of unselected ones as the process requestors to be
3 inserted and read, inserting said message into the message queues related to the process
4 requestors to be inserted and read, and reading the agents related to said process
5 requestors from the persistent storage to the cache memory;

6 an agent instruction step of instructing the agent related to the message queue having the
7 message inserted therein to operate; and

8 a repetitive instruction step of, in the case where the unselected one remains among the
9 process requestors included in said list information, waiting for termination of the process
10 of all the agents in operation and instructing said insertion and reading step to be
11 repeated.

12 (17) The message processing method according to claim 16, characterized in that each
13 agent sends a notice to the process requestor associated with the agent as to the message
14 in the message queue associated with the agent.

15 (18) A message processing method characterized by having:

16 a process requestor search information management step of managing process requestor
17 search information for searching for an applicable process requestor as to an agent start
18 cause event;

19 an acceptance step of accepting the agent start cause event;

1 a list information creation step of, based on said process requestor search information,
2 creating list information on process requestors to which a message generated due to said
3 agent start cause event is applied;

4 an agent setting step of setting a plurality of agents, the agents being associated with the
5 process requestors, stored in a persistent storage, readable from the persistent storage to a
6 cache memory as a program execution area and abandonable from the cache memory,
7 each agent operating only when existing in the cache memory to be able to process the
8 message in a message queue corresponding to the agent;

9 a first message queue processing step;

10 a second message queue processing step;

11 a selection step of selecting either one of the first and second message queue processing
12 steps; and

13 an agent instruction step of, as to the agent related to the message queue having the
14 message inserted therein, immediately instructing the agent to operate if the agent is in
15 the cache memory, and reading the agent from said persistent storage to said cache
16 memory and then instructing the agent to operate if the agent is not in the cache memory;
17 and wherein:

18 the first message queue processing step has an insertion step of inserting said message
19 into the message queues related to all the process requestors included in said list
20 information; and

21 the second message queue processing step has:

1 an insertion and reading step of, of the process requestors included in said list
2 information, selecting a plurality of unselected ones as the process requestors to be
3 inserted and read, inserting said message into the message queues related to the process
4 requestors to be inserted and read and reading the agents related to said process requestors
5 from the persistent storage to the cache memory; and

6 a repetitive instruction step of, in the case where the unselected one remains among the
7 process requestors included in said list information, waiting for termination of the process
8 of all the agents in operation and instructing said insertion and reading step to be
9 repeated.

10 (19) The message processing method according to claim 18, characterized in that the
11 selection in said selection step is based on an instruction of an operator.

12 (20) The message processing method according to claim 18, characterized in that the
13 selection in said selection step is based on the estimated number of the process requestors
14 to which the message generated due to the agent start cause event of this time is applied,
15 estimated hit rate in the cache memory as to the agent related to the process requestors to
16 which the message generated due to the agent start cause event of this time is applied,
17 estimated work time, in the case where the process is allocated in the first message queue
18 processing step, from acceptance of the information in said acceptance step until
19 obtaining the list information on the process requestors to which the message is applied
20 and inserting the message into the message queues of all the agents, estimated work time,
21 in the case where the process is allocated in the second message queue processing step,
22 from acceptance of the information in said acceptance step until obtaining the list
23 information on the process requestors to which the message is applied and inserting the

1 message into the message queues of all the agents, estimated time from determination of
2 use as to the agent in the cache memory until completion of the process by the determined
3 agent, and/or estimated time from determination of use as to the agent outside the cache
4 memory until the completion of the process by the determined agent.

5 (21) A message processing method characterized by having:

6 a process requestor search information management step of managing process requestor
7 search information for searching for an applicable process requestor as to an agent start
8 cause event;

9 an acceptance step of accepting the agent start cause event;

10 a process requestor determination step of determining the process requestor to which the
11 message generated due to said agent start cause event is applied based on said process
12 requestor search information;

13 an agent setting step of setting a plurality of agents, the agents being associated with the
14 process requestors, stored in a persistent storage, each agent being readable from the
15 persistent storage to a cache memory as a program execution area and abandonable from
16 the cache memory, and each agent operating only when existing in the cache memory to
17 be able to process the message in a message queue corresponding to the agent;

18 at least one sub-process priority determination step of determining process priority about
19 each message as sub-process priority based on a single standard of value;

20 a compound process priority determination step of, when the total number of said

1 sub-process priority determination steps is two or more, determining compound process
2 priority about each message based on the sub-process priority individually determined as
3 to each message in each sub-process priority determination step, and when the total
4 number of said sub-process priority determination steps is one, determining as the
5 compound process priority the sub-process priority determined in said one sub-process
6 priority determination step as to each message; and

7 an agent instruction step of rendering the message of the highest compound process
8 priority among the messages held by each message queue as the message of the highest
9 priority, and between the agents related to the message queues of which compound
10 process priority of the message of the highest priority is the same, instructing the agent
11 existing in the cache memory to operate in preference to the agent not existing therein.

12 (22) The message processing method according to claim 21, characterized in that said
13 standard of value may be related to the contents of the message or to the process
14 requestors to which the message is applied.

15 (23) The message processing method according to claim 22, characterized in that the
16 predetermined standard of value related to the contents of the message includes the
17 standard of value related to emergency of processing of the message.

18 (24) The message processing method according to claim 22, characterized in that the
19 standard of value related to the process requestors to which the message is applied
20 includes the standard of value related to rating of the process requestors.

21 (25) The message processing method according to claim 21, characterized in that, when
22 there are a plurality of messages held by the message queue related to the agent which

1 once started the process in said agent instruction step, the agent continuously processes
2 all those messages or the messages of which compound process priority is within a
3 predetermined position in descending rank.

4 (26) The message processing method according to claim 21, characterized in that:

5 there is an agent management step including said sub-process priority determination step
6 and said compound process priority determination step;

7 said agent management step has an existence detection step of detecting whether or not
8 each agent exists in the persistent storage, and grouping information management step of
9 grouping the agents, managing grouping information and updating the grouping
10 information as appropriate based on determination results of the existence detection step
11 and the compound process priority of each agent; and
12 said agent instruction step instructs the agents to operate in order based on the grouping
13 information in said agent management step.

14 (27) A message processing method characterized by having:

15 a process requestor search information management step of managing process requestor
16 search information for searching for an applicable process requestor as to an agent start
17 cause event;

18 an acceptance step of accepting the agent start cause event;

19 an acceptance order information management step of managing acceptance order
20 information on the agent start cause events accepted by said acceptance step;

1 an agent setting step of setting a plurality of agents, the agents being associated with the
2 process requestors, stored in a persistent storage, readable from the persistent storage to a
3 cache memory as a program execution area and abandonable from the cache memory, and
4 each agent operating only when existing in the cache memory to be able to process the
5 message in a message queue corresponding to the agent;

6 a thread setting step of setting a plurality of threads, the threads being mutually operable
7 in parallel, detecting the process requestors to which the message generated due to the
8 agent start cause event is applied based on said process requestor search information and
9 inserting said message into the message queues related to the process requestors;

10 an allocation step of allocating to each thread the agent start cause event to be processed
11 by that thread;

12 a proceeding information management step of managing proceeding information on the
13 process by each thread as to each agent start cause event accepted by said acceptance step;

14 a determination step of, as to the agent start cause event of which process proceeding
15 information is the information on thread process termination (hereafter, referred to as a
16 “determined agent start cause event”), determining whether or not, of the agent start cause
17 events accepted in said acceptance step prior to the determined agent start cause event,
18 there is any agent start cause event of which thread process is unfinished; and

19 an agent control step of controlling the process by the agent as to the message generated
20 due to the determined agent start cause event determined as “yes” in said determination
21 step.

- 1 (28) The message processing method according to claim 27, characterized by having
2 said agent control step of allowing the process by the agent as to the message generated
3 due to the determined agent start cause event determined as “no” in said determination
4 step.
- 5 (29) The message processing method according to claim 27, characterized by having
6 said determination step of, in the case where the agent start cause event immediately
7 following the agent start cause event determined as “no” in acceptance order is already
8 determined as “yes,” changing the determination result from “yes” to “no.”
- 9 (30) The message processing method according to claim 29, characterized by having
10 said agent setting step of, in the case of causing the agent to process the message queue in
11 which the messages generated due to a plurality of agent start cause events of which
12 determination result by said determination step is “no” continue in an acceptance order
13 direction, setting said agent to continuously process the plurality of continuous messages.
- 14 (31) A message processing program characterized by causing a computer to execute the
15 steps in any one of the message processing methods according to claims 16 to 30.
- 16 (32) A computer program product comprising a computer usable medium having
17 computer readable program code means embodied therein for causing message
18 processing, the computer readable program code means in said computer program
19 product comprising computer readable program code means for causing a computer to
20 effect the functions of claim 1.
- 21 (33) A computer program product comprising a computer usable medium having

1 computer readable program code means embodied therein for causing message
2 processing, the computer readable program code means in said computer program
3 product comprising computer readable program code means for causing a computer to
4 effect the functions of claim 3.

5 (34) A computer program product comprising a computer usable medium having
6 computer readable program code means embodied therein for causing message
7 processing, the computer readable program code means in said computer program
8 product comprising computer readable program code means for causing a computer to
9 effect the functions of claim 6.

10 (35) A computer program product comprising a computer usable medium having
11 computer readable program code means embodied therein for causing message
12 processing, the computer readable program code means in said computer program
13 product comprising computer readable program code means for causing a computer to
14 effect the functions of claim 12.

15 (36) An article of manufacture comprising a computer usable medium having computer
16 readable program code means embodied therein for causing message processing, the
17 computer readable program code means in said article of manufacture comprising
18 computer readable program code means for causing a computer to effect the steps of
19 claim 16.

20 (37) An article of manufacture comprising a computer usable medium having computer
21 readable program code means embodied therein for causing message processing, the
22 computer readable program code means in said article of manufacture comprising
23 computer readable program code means for causing a computer to effect the steps of

1 claim 18.

2 (38) An article of manufacture comprising a computer usable medium having computer
3 readable program code means embodied therein for causing message processing, the
4 computer readable program code means in said article of manufacture comprising
5 computer readable program code means for causing a computer to effect the steps of
6 claim 21.

7 (39) An article of manufacture comprising a computer usable medium having computer
8 readable program code means embodied therein for causing message processing, the
9 computer readable program code means in said article of manufacture comprising
10 computer readable program code means for causing a computer to effect the steps of
11 claim 27.

12 (40) A program storage device readable by machine, tangibly embodying a program of
13 instructions executable by the machine to perform method steps for message processing,
14 said method steps comprising the steps of claim 16.

15 (41) A program storage device readable by machine, tangibly embodying a program of
16 instructions executable by the machine to perform method steps for message processing,
17 said method steps comprising the steps of claim 18.

18 (42) A program storage device readable by machine, tangibly embodying a program of
19 instructions executable by the machine to perform method steps for message processing,
20 said method steps comprising the steps of claim 21.

21 (43) A program storage device readable by machine, tangibly embodying a program of

- 1 instructions executable by the machine to perform method steps for message processing,
- 2 said method steps comprising the steps of claim 27.